



## **CUPS Interface Design Description**

CUPS-IDD-1.1

Easy Software Products  
Copyright 1997–2001, All Rights Reserved



# Table of Contents

<b><u>1 Scope</u></b>	<b><u>1</u></b>
<u>1.1 Identification</u>	<u>1</u>
<u>1.2 System Overview</u>	<u>1</u>
<u>1.3 Document Overview</u>	<u>1</u>
<b><u>2 References</u></b>	<b><u>3</u></b>
<u>2.1 CUPS Documentation</u>	<u>3</u>
<u>2.2 Other Documents</u>	<u>3</u>
<b><u>3 Internal Interfaces</u></b>	<b><u>5</u></b>
<u>3.1 Character Set Files</u>	<u>5</u>
<u>3.1.1 8-Bit Character Set Files</u>	<u>5</u>
<u>3.1.2 Unicode Character Set Files</u>	<u>5</u>
<u>3.2 Language Files</u>	<u>6</u>
<u>3.3 MIME Files</u>	<u>7</u>
<u>3.3.1 mime.types</u>	<u>7</u>
<u>3.3.2 mime.convs</u>	<u>7</u>
<u>3.4 Option Files</u>	<u>7</u>
<u>3.5 PostScript Printer Description Files</u>	<u>8</u>
<u>3.5.1 PPD Specification</u>	<u>8</u>
<u>3.5.2 CUPS Extensions to PPD Files</u>	<u>8</u>
<u>3.6 Scheduler Configuration Files</u>	<u>9</u>
<u>3.6.1 classes.conf</u>	<u>9</u>
<u>3.6.2 cupsd.conf</u>	<u>10</u>
<u>3.6.3 printers.conf</u>	<u>14</u>
<b><u>4 External Interfaces</u></b>	<b><u>17</u></b>
<u>4.1 AppSocket Protocol</u>	<u>17</u>
<u>4.2 CUPS Browsing Protocol</u>	<u>17</u>
<u>4.3 CUPS Form File</u>	<u>18</u>
<u>4.3.1 CUPS Form DTD</u>	<u>18</u>
<u>4.4 CUPS PostScript File</u>	<u>21</u>
<u>4.5 CUPS Raster File</u>	<u>21</u>
<u>4.6 CUPS Raw Files</u>	<u>23</u>
<u>4.7 Internet Printing Protocol</u>	<u>24</u>
<u>4.8 Line Printer Daemon Protocol</u>	<u>24</u>
<u>4.9 Server Message Block Protocol</u>	<u>24</u>
<b><u>5 Directories</u></b>	<b><u>25</u></b>
<b><u>A Glossary</u></b>	<b><u>27</u></b>
<u>A.1 Terms</u>	<u>27</u>
<u>A.2 Acronyms</u>	<u>27</u>



# 1 Scope

## 1.1 Identification

This interface design description document provides detailed file formats, message formats, and program conventions for the Common UNIX Printing System ("CUPS") Version 1.1.

## 1.2 System Overview

CUPS provides a portable printing layer for UNIX®-based operating systems. It has been developed by [Easy Software Products](#) to promote a standard printing solution for all UNIX vendors and users. CUPS provides the System V and Berkeley command-line interfaces.

CUPS uses the Internet Printing Protocol ("IPP") as the basis for managing print jobs and queues. The Line Printer Daemon ("LPD") Server Message Block ("SMB"), and AppSocket (a.k.a. JetDirect) protocols are also supported with reduced functionality. CUPS adds network printer browsing and PostScript Printer Description ("PPD") based printing options to support real-world printing under UNIX.

CUPS also includes a customized version of GNU Ghostscript (currently based off GNU Ghostscript 5.50) and an image file RIP that are used to support non-PostScript printers. Sample drivers for HP and EPSON printers are included that use these filters.

## 1.3 Document Overview

This interface design description document is organized into the following sections:

- 1 – Scope
- 2 – References
- 3 – Internal Interfaces
- 4 – External Interfaces
- 5 – Directories
- A – Glossary



## 2 References

### 2.1 CUPS Documentation

The following CUPS documentation is referenced by this document:

- CUPS–CMP–1.1: CUPS Configuration Management Plan
- CUPS–IDD–1.1: CUPS System Interface Design Description
- CUPS–IPP–1.1: CUPS Implementation of IPP
- CUPS–SAM–1.1.x: CUPS Software Administrators Manual
- CUPS–SDD–1.1: CUPS Software Design Description
- CUPS–SPM–1.1.x: CUPS Software Programming Manual
- CUPS–SSR–1.1: CUPS Software Security Report
- CUPS–STP–1.1: CUPS Software Test Plan
- CUPS–SUM–1.1.x: CUPS Software Users Manual
- CUPS–SVD–1.1: CUPS Software Version Description

### 2.2 Other Documents

The following non–CUPS documents are referenced by this document:

- [Adobe PostScript Printer Description File Format Specification, Version 4.3.](#)
- [Adobe PostScript Language Reference, Third Edition.](#)
- IPP: Job and Printer Set Operations
- IPP/1.1: Encoding and Transport
- IPP/1.1: Implementers Guide
- IPP/1.1: Model and Semantics
- [RFC 1179, Line Printer Daemon Protocol](#)
- [RFC 2567, Design Goals for an Internet Printing Protocol](#)
- [RFC 2568, Rationale for the Structure of the Model and Protocol](#) for the Internet Printing Protocol
- [RFC 2569, Mapping between LPD and IPP Protocols](#)
- [RFC 2616, Hypertext Transfer Protocol — HTTP/1.1](#)
- [RFC 2617, HTTP Authentication: Basic and Digest Access](#) Authentication





## 3 Internal Interfaces

### 3.1 Character Set Files

The character set files define a mapping between 8-bit characters and the Unicode character set, or between Unicode and printer fonts. They are named using the IETF charset names defined in RFCnnnn. These files are ASCII text, the content of which is described below. Comments can be included by using the # character in the first column of a line.

#### 3.1.1 8-Bit Character Set Files

8-bit character set files start with a line reading:

```
charset 8bit
```

Following this are lines that define the font information:

```
first last direction width normal bold italic bold-italic
```

*First* and *last* are the first and last glyphs in the font mapping that correspond to that font; a maximum of 256 characters can be mapped within each group, with a maximum of 256 mappings (this is a PostScript limitation.) The glyph values are hexadecimal.

*Direction* is the string "ltor", "rtol", or "rtola" indicating left-to-right, right-to-left, or right-to-left Arabic text.

*Width* is the string "single" or "double"; double means that the glyphs are twice as wide as ASCII characters in the Courier typeface.

*Normal*, *bold*, *italic*, and *bold-italic* are the typefaces to use for each presentation. If characters are only available in a single style then only one typeface should be listed (e.g. "Symbol".) Each font that is listed will be used (and downloaded if needed) when printing.

The remaining lines define a character to Unicode glyph mapping for the character set. The character and glyph values are hexadecimal:

```
xx yyyy
```

#### 3.1.2 Unicode Character Set Files

Unicode character set files start with a line reading:

```
charset encoding
```

*Encoding* is the encoding to use for the text; currently only the string "utf8" is supported.

Following this are lines defining the font information:

```
first last direction width normal bold italic bold-italic
```

*First* and *last* are the first and last glyphs in the font mapping that correspond to that font; a maximum of 256 characters can be mapped within each group, with a maximum of 256 mappings (this is a PostScript limitation.) The glyph values are hexadecimal.

*Direction* is the string "ltor", "rtol", or "rtola" indicating left-to-right, right-to-left, or right-to-left Arabic text.

*Width* is the string "single" or "double"; double means that the glyphs are twice as wide as ASCII characters in the Courier typeface.

*Normal*, *bold*, *italic*, and *bold-italic* are the typefaces to use for each presentation. If characters are only available in a single style then only one typeface should be listed (e.g. "Symbol".) Each font that is listed will be used (and downloaded if needed) when printing.

## 3.2 Language Files

The language files define the default character set and a collection of text messages in that language. They are named by prefixing the string "cups\_" to the front of the language specifier (e.g. "cups\_en", "cups\_fr", etc.) Each file consists of two or more lines of ASCII text.

The first line identifies the character set to be used for the messages. The currently recognized values are:

- iso-8859-1
- iso-8859-2
- iso-8859-3
- iso-8859-4
- iso-8859-5
- iso-8859-6
- iso-8859-7
- iso-8859-8
- iso-8859-9
- iso-8859-10
- iso-8859-13
- iso-8859-14
- iso-8859-15
- us-ascii
- utf-8
- windows-874
- windows-1250
- windows-1251
- windows-1252
- windows-1253
- windows-1254
- windows-1255
- windows-1256
- windows-1257
- windows-1258

The second and succeeding lines define text messages. If the message text is preceded by a number, then the current message number is updated and the text after the number is used.

## 3.3 MIME Files

CUPS uses two MIME files in its standard configuration.

### 3.3.1 mime.types

The mime.types file defines the recognized file types and consists of 1 or more lines of ASCII text. Comment lines start with the pound ("#") character. The backslash ("\") character can be used at the end of a line to continue that line to the next.

Each non-blank line starts with a MIME type identifier ("super/type") as registered with the IANA. All text following the MIME type is treated as a series of type recognition rules:

```
mime-type := super "/" type { SP rule }*
super := { "a-z" | "A-Z" }*
type := { "a-z" | "A-Z" | "-" | "." | "0-9" }*
rule := { extension | match | operator | "(" rule ")" }*
extension := { "a-z" | "A-Z" | "0-9" }*
match := "match(" regexp ")" |
         "ascii(" offset "," length ")" |
         "printable(" offset "," length ")" |
         "string(" offset "," string ")" |
         "contains(" offset "," length "," string ")" |
         "char(" offset "," value ")" |
         "short(" offset "," value ")" |
         "int(" offset "," value ")" |
         "locale(" string ")"
operator := "+" | [ logical AND ]
           "," | SP [ logical OR ]
           "!" | [ unary NOT ]
```

The int and short rules match look for integers in network byte order (a.k.a. big-endian) with the most-significant byte first.

### 3.3.2 mime.convs

The mime.types file defines the recognized file filters and consists of 1 or more lines of ASCII text. Comment lines start with the pound ("#") character.

Each non-blank line starts with two MIME type identifiers ("super/type") representing the source and destination types. Following the MIME types are a cost value (0 to 100) and the filter program to use. If the filter program is not specified using the full path then it must reside in the CUPS filter directory:

```
super/type SP super/type2 SP cost SP program
```

## 3.4 Option Files

CUPS maintains user-defined printer and option files for each printer and user on the system. The printers and options defined in the system option file (/etc/cups/lpoptions) are loaded first, followed by the user option file (\$HOME/.lpoptions). Options in the user file replace those defined in the system file for the same destination. Each line in the files can be one of the following:

```
Dest name option=value option=value ... option=value
```

```
Dest name/instance option=value option=value ... option=value
Default name option=value option=value ... option=value
Default name/instance option=value option=value ... option=value
```

The line beginning with "Default" indicates the default destination for print jobs; a default line in the user option file overrides the default defined in the system option file.

*Name* is the name of a printer known to the local server.

*Instance* can be any string of letters, numbers, and the underscore up to 127 characters in length.

The remainder of the line contains a list of space-separated options and their values.

## 3.5 PostScript Printer Description Files

PostScript Printer Description ("PPD") files describe the capabilities of each printer and are used by CUPS to support printer-specific features and intelligent filtering.

### 3.5.1 PPD Specification

The PPD file format is described in [Adobe TechNote #5003: PostScript Printer Description File Format Specification Version 4.3](#).

### 3.5.2 CUPS Extensions to PPD Files

CUPS adds several new attributes that are described below.

#### 3.5.2.1 cupsFilter

This string attribute provides a conversion rule of the form:

```
source/type cost program
```

The destination type is assumed to be the printer's type. If a printer supports the source type directly the special filter program "-" may be specified.

#### 3.5.2.2 cupsManualCopies

This boolean attribute notifies the RIP filters that the destination printer does not support copy generation in hardware. The default value is false.

#### 3.5.2.3 cupsModelNumber

This integer attribute specifies a printer-specific model number. This number can be used by a filter program to adjust the output for a specific model of printer.

#### 3.5.2.4 cupsProfile

This string attribute specifies a color profile of the form:

```
resolution/type density gamma m00 m01 m02 m10 m11 m12 m20 m21 m22
```

The *resolution* and *type* values may be "-" to act as a wildcard. Otherwise they must match one of the *Resolution* or *MediaType* attributes defined in the PPD file.

The *density* and *gamma* values define gamma and density adjustment function such that:

$$f(x) = \text{density} * x\text{gamma}$$

The *m00* through *m22* values define a 3x3 transformation matrix for the CMY color values. The density function is applied *after* the CMY transformation.

### 3.5.2.5 cupsVersion

This required attribute describes which version of the CUPS IDD was used for the PPD file extensions. Currently it must be the string "1.0" or "1.1".

## 3.6 Scheduler Configuration Files

The scheduler reads three configuration files that define the available printers, classes, and services:

*classes.conf*

This file defines all of the printer classes known to the system.

*cupsd.conf*

This file defines the files, directories, passwords, etc. used by the scheduler.

*printers.conf*

This file defines all of the printers known to the system.

### 3.6.1 classes.conf

The *classes.conf* file consists of 1 or more lines of ASCII text. Comment lines start with the pound ("#") character.

Each non-blank line starts with the name of a configuration directive followed by its value. The following directives are understood:

Directive	Description
<Class name> </Class>	Surrounds a class definition.
<DefaultClass name> </Class>	Surrounds a class definition for the default destination.
Accepting	Specifies whether the class is accepting new jobs. May be the names "Yes" or "No".
AllowUsers	Specifies a list of users that are allowed to access the class.
BannerStart	Specifies the banner that is printed before other files in a job.
BannerEnd	

	Specifies the banner that is printed after other files in a job.
DenyUsers	Specifies a list of users that are not allowed to access the class.
Info	A textual description of the class.
Location	A textual location of the class.
Printer	Specifies a printer that is a member of the class.
State	Specifies the initial state of the class; can be "Idle" or "Stopped".
StateMessage	Specifies a textual message for the current class state.

### 3.6.2 cupsd.conf

The cupsd.conf file consists of 1 or more lines of ASCII text. Comment lines start with the pound ("#") character.

Each non-blank line starts with the name of a configuration directive followed by its value. The following directives are understood:

Directive	Default	Description
AccessLog	access_log	Specifies the location of the access log file. The special name "syslog" can be used to send access log information to the system log.
Allow	—	Allows connections from the specified host, network, or domain.
AuthClass	—	Specifies what level of authentication is required; may be "User", "System", or "Group".
AuthType	None	Specifies the type of authentication to perform; may be "None", "Basic", or "Digest".
BrowseAddress	255.255.255.255	Specifies a broadcast address to send CUPS browsing packets to.
BrowseAllow	—	Specifies hosts or addresses from which

## CUPS Interface Design Description

		browsing information should be used.
BrowseDeny	—	Specifies hosts or addresses from which browsing information should not be used.
BrowseInterval	30	Specifies the number of seconds between browsing updates. A browse interval of 0 seconds disables outgoing packets.
BrowseOrder	Allow,Deny	Specifies the order of BrowseAllow and BrowseDeny directive processing; can be "Deny,Allow" to implicitly deny hosts unless they are allowed by a BrowseAllow line, or "Allow,Deny" to implicitly allow hosts unless they are denied by a BrowseDeny line.
BrowsePoll	—	Specifies a server to poll for available printers and classes.
BrowsePort	631	Specifies the UDP port number to use for browse packets.
BrowseRelay	—	Specifies a source and destination address for relaying browser information from one subnet to another.
BrowseShortNames	yes	Specifies whether or not to provide short names (without the "@server" part) for remote printers.
BrowseTimeout	300	Specifies the number of seconds to wait until remote destinations are removed from the local destination list.
Browsing	On	Specifies whether or not printer and class browsing is enabled; can be "On" or

## CUPS Interface Design Description

		"Off".
DataDir	/usr/share/cups	Specifies the directory where CUPS data files are stored.
DefaultCharset	iso-8859-1	Specifies the default character set.
DefaultLanguage	current locale	Specifies the default language.
Deny	—	Refuses connections from the specified host, network, or domain.
DocumentRoot	/usr/share/doc/cups	Specifies the document data root directory.
ErrorLog	error_log	Specifies the error log file location. The special name "syslog" can be used to send error log information to the system log.
Group	root, sys, system	Specifies the group name or ID that is used when running external programs.
HostNameLookups	Off	Specifies whether or not to perform reverse IP address lookups to get the actual hostname; may be "On" or "Off". Hostname lookups can significantly degrade the performance of the CUPS server if one or more DNS servers is not functioning properly.
ImplicitClasses	On	Specifies whether or not to automatically create printer classes when more than one printer or class of the same name is detected on the network; may be "On" or "Off".
KeepAlive	On	Specifies whether or not to use the HTTP Keep-Alive feature; may be "On" or "Off".
KeepAliveTimeout	30	Specifies the amount of time to keep the HTTP connection alive before



## CUPS Interface Design Description

		closing it.
<Location path> </Location>	—	Specifies a location to restrict access to.
LogLevel	info	Controls the amount of information that is logged in the error log file. Can be one of "debug", "info", "warn", "error", or "none", in decreasing order or verbosity.
MaxClients	100	Specifies the maximum number of simultaneous active clients. This value is internally limited to 1/3 of the total number of available file descriptors.
MaxLogSize	0	Specifies the maximum size of the access, error, and page log files in bytes. If set to 0 then no maximum size is set. Log files are rotated automatically when this size is exceeded.
MaxRequestSize	0	Specifies the maximum size of HTTP requests in bytes. If set to 0 then there is no maximum.
Order	Allow,Deny	Specifies the order of Allow and Deny directive processing; can be "Deny,Allow" to implicitly deny hosts unless they are allowed by an Allow line, or "Allow,Deny" to implicitly allow hosts unless they are denied by a Deny line.
PageLog	page_log	Specifies the location of the page log file. The special name "syslog" can be used to send page log information to the system log.
Port	631	Specifies a port number to listen to for HTTP

		connections.
Printcap	/etc/printcap	Specifies the location of a Berkeley printcap file to update with a list of current printers and classes. If no filename is supplied then this automatic generation is disabled.
RequestRoot	/var/spool/cups	Specifies the location of request files.
RIPCache	8m	Specifies the size of the memory cache in bytes that is used by RIP filters.
ServerAdmin	root@ServerName	Specifies the person to contact with problems.
ServerName	hostname	Specifies the hostname that is supplied to HTTP clients. This is also used to determine the default CUPS server for the CUPS IPP client applications.
ServerRoot	/etc/cups	Specifies the root directory for server configuration files.
SystemGroup	root, sys, system	Specifies the group name used for System class authentication.
TempDir	/var/tmp	Specifies the temporary directory to use.
Timeout	300	The timeout in seconds before client connections are closed in the middle of a request.
User	lp	Specifies the user that is used when running external programs.

### 3.6.3 printers.conf

The printers.conf file consists of 1 or more lines of ASCII text. Comment lines start with the pound ("#") character.

Each non-blank line starts with the name of a configuration directive followed by its value. The following

directives are understood:

Directive	Description
Accepting	Specifies whether the printer is accepting new jobs. May be the names "Yes" or "No".
<DefaultPrinter name> </Printer>	Surrounds the printer definition for a default destination.
AllowUsers	Specifies a list of users that are allowed to access the printer.
BannerStart	Specifies the banner that is printed before other files in a job.
BannerEnd	Specifies the banner that is printed after other files in a job.
DenyUsers	Specifies a list of users that are not allowed to access the printer.
DeviceURI	Specifies the device-uri attribute for the printer.
Info	A textual description of the printer.
Location	A textual location of the printer.
<Printer name> </Printer>	Surrounds the printer definition.
State	Specifies the initial state of the printer; can be "Idle" or "Stopped".
StateMessage	Specifies a textual message for the current printer state.



## 4 External Interfaces

### 4.1 AppSocket Protocol

The AppSocket protocol is an 8-bit clean TCP/IP socket connection. The default IP service port is 9100. The URI method name is "socket".

The AppSocket protocol is used by the Hewlett Packard JetDirect network interfaces and print servers, as well as many other vendors' products. See the CUPS Software Administrators Manual for a list of supported products.

### 4.2 CUPS Browsing Protocol

The CUPS Browsing Protocol is a UDP/IP-based broadcast service. By default this service operates on IP service port 631.

Each broadcast packet describes the state of a single printer or class and is an ASCII text string of up to 1450 bytes ending with a newline (0x0a). The string is formatted as follows:

```
type SP state SP uri SP "location" SP "info" SP "make-and-model" NL
```

*State*, *uri*, *location*, *info*, and *make-and-model*, correspond to the IPP *printer-state*, *printer-uri-supported*, *printer-location*, *printer-info*, and *printer-make-and-model* attributes.

*Type* is a hexadecimal number string representing capability/type bits:

Bit	Description
0	0 = printer 1 = class
1	0 = local 1 = remote (always 1)
2	1 = can print B
3	1 = can print color
4	1 = can duplex
5	1 = can staple
6	1 = can do fast copies
7	1 = can do fast collating
8	1 = can punch holes
9	1 = can cover
10	1 = can bind
11	1 = can sort

12	1 = can print up to 9x14 inches
13	1 = can print up to 18x24 inches
14	1 = can print up to 36x48 inches
15	1 = can print variable sizes

## 4.3 CUPS Form File

CUPS Form files are XML files used by the CUPS `formtops` filter to produce dynamic banner pages and support preprinted forms.

The MIME type for CUPS Form files is `application/vnd.cups-form`.

### 4.3.1 CUPS Form DTD

The following DTD describes the available elements and attributes in a CUPS Form file:

```
<!ENTITY % Angle "CDATA" -- angle in degrees -->

<!ENTITY % Color "CDATA" -- a color using sRGB: #RRGGBB as Hex values -->

<!ENTITY % Length "CDATA" -- nn for pixels or nn% for percentage length -->

<!ENTITY % Lengths "CDATA" -- comma-separated Length values -->

<!ENTITY % Text "CDATA">

<!ENTITY % heading "H1|H2|H3|H4|H5|H6">

<!ENTITY % preformatted "PRE">

<!ENTITY % i18n
  "lang      %LanguageCode; #IMPLIED  -- language code --
  dir        (ltr|rtl)      #IMPLIED  -- direction for weak/neutral text --"
>

<!ENTITY % attrs "%i18n;">

<!ENTITY % fontstyle
  "B | FONT | I | TT">

<!ENTITY % graphics
  "BOX | RECT | LINE | POLY | ARC | PIE | TEXT">

<!ENTITY % insert
  "IMG | VAR">

<!-- %inline; covers inline or "text-level" elements -->
<!ENTITY % inline "#PCDATA | %fontstyle; | %graphics; | %insert;">

<!ELEMENT (%fontstyle;) - - (%inline;)*>
```

## CUPS Interface Design Description

```

<!ATTLIST (%fontstyle;)
    %attrs;                                -- %i18n --
>

<!ELEMENT BR - O EMPTY                    -- forced line break -->
<!ATTLIST BR
    %attrs;                                -- %i18n --
>

<!ENTITY % block
    "P | %heading; | %preformatted;">

<!ENTITY % flow "%block; | %inline;">

<!ELEMENT PAGE O O (%flow;)+              -- document body -->
<!ATTLIST PAGE
    %attrs;                                -- %i18n --
    align      (left|center|right) #IMPLIED -- horizontal alignment --
    valign     (top|middle|center|bottom) #IMPLIED -- vertical alignment --
>

<!ELEMENT P - O (%inline;)*              -- paragraph -->
<!ATTLIST P
    %attrs;                                -- %i18n --
    align      (left|center|right) #IMPLIED -- horizontal alignment --
>

<!ELEMENT (%heading;) - - (%inline;)* -- heading -->
<!ATTLIST (%heading;)
    %attrs;                                -- %i18n --
    align      (left|center|right) #IMPLIED -- horizontal alignment --
>

<!ELEMENT PRE - - (%inline;)*            -- preformatted text -->
<!ATTLIST PRE
    %attrs;                                -- %i18n --
    align      (left|center|right) #IMPLIED -- horizontal alignment --
>

<!ELEMENT BOX - O EMPTY                  -- unfilled box -->
<!ATTLIST BOX
    color      %Color;                    #IMPLIED -- override color --
    height     %Length;                   #REQUIRED -- height of box --
    thickness   %Length;                   #IMPLIED -- override line thickness --
    width      %Length;                   #REQUIRED -- width of box --
    x          %Length;                   #REQUIRED -- horizontal position --
    y          %Length;                   #REQUIRED -- vertical position --
>

<!ELEMENT RECT - O EMPTY                 -- filled box -->
<!ATTLIST RECT
    color      %Color;                    #IMPLIED -- override color --
    height     %Length;                   #REQUIRED -- height of box --
    width      %Length;                   #REQUIRED -- width of box --
    x          %Length;                   #REQUIRED -- horizontal position --
    y          %Length;                   #REQUIRED -- vertical position --
>

<!ELEMENT LINE - O EMPTY                 -- polyline -->
<!ATTLIST LINE
    color      %Color;                    #IMPLIED -- override color --
    thickness   %Length;                   #IMPLIED -- override line thickness --

```

## CUPS Interface Design Description

```

x          %Lengths;      #REQUIRED -- horizontal positions --
y          %Lengths;      #REQUIRED -- vertical positions --
>

<!ELEMENT POLY - O EMPTY          -- polygon (filled) -->
<!ATTLIST POLY
  color      %Color;          #IMPLIED -- override color --
  x          %Lengths;        #REQUIRED -- horizontal positions --
  y          %Lengths;        #REQUIRED -- vertical positions --
>

<!ELEMENT ARC - O EMPTY          -- unfilled arc -->
<!ATTLIST ARC
  color      %Color;          #IMPLIED -- override color --
  end        %Angle;          #IMPLIED -- override end angle --
  height     %Length;         #REQUIRED -- height of arc --
  start      %Angle;          #IMPLIED -- override start angle --
  thickness  %Length;         #IMPLIED -- override line thickness --
  width      %Length;         #REQUIRED -- width of arc --
  x          %Length;         #REQUIRED -- horizontal position --
  y          %Length;         #REQUIRED -- vertical position --
>

<!ELEMENT PIE - O EMPTY          -- filled arc -->
<!ATTLIST PIE
  color      %Color;          #IMPLIED -- override color --
  end        %Angle;          #IMPLIED -- override end angle --
  height     %Length;         #REQUIRED -- height of arc --
  start      %Angle;          #IMPLIED -- override start angle --
  width      %Length;         #REQUIRED -- width of arc --
  x          %Length;         #REQUIRED -- horizontal position --
  y          %Length;         #REQUIRED -- vertical position --
>

<!ELEMENT TEXT - - (%flow;)*      -- text box -->
<!ATTLIST RECT
  align      (left|center|right) #IMPLIED -- horizontal alignment --
  height     %Length;         #REQUIRED -- height of box --
  valign     (top|middle|center|bottom) #IMPLIED -- vertical alignment --
  width      %Length;         #REQUIRED -- width of box --
  x          %Length;         #REQUIRED -- horizontal position --
  y          %Length;         #REQUIRED -- vertical position --
>

<!ELEMENT IMG - O EMPTY          -- Embedded image -->
<!ATTLIST IMG
  %attrs;    -- %coreattrs, %il8n, %events --
  src        %URI;           #REQUIRED -- URI of image to embed --
  height     %Length;        #IMPLIED -- override height --
  width      %Length;        #IMPLIED -- override width --
>

<!ELEMENT HEAD O O (DEFVAR)*      -- document head -->
<!ATTLIST HEAD
  %il8n;     -- lang, dir --
>

<!ELEMENT DEFVAR - O EMPTY        -- variable definition -->
<!ATTLIST DEFVAR
  name       CDATA           #REQUIRED -- name
  value      CDATA           #REQUIRED -- value

```



```

>

<!ENTITY % html.content "HEAD, PAGE">

<!ELEMENT CUPSFORM - - (HEAD) (PAGE)+ -- document root element -->
<!ATTLIST CUPSFORM
    %il8n;                                -- lang, dir --
>

```

## 4.4 CUPS PostScript File

CUPS PostScript files are device-dependent Adobe PostScript program files. The PostScript language is described in the [Adobe PostScript Language Reference Manual, Third Edition](#).

The MIME type for CUPS PostScript files is `application/vnd.cups-postscript`.

## 4.5 CUPS Raster File

CUPS raster files are device-dependent raster image files that contain a PostScript page device dictionary and device-dependent raster imagery for each page in the document. These files are used to transfer raster data from the PostScript and image file RIPs to device-dependent filters that convert the raster data to a printable format.

A raster file begins with a four byte synchronization word: 0x52615374 ("RaSt") for big-endian architectures and 0x74536152 ("tSaR") for little-endian architectures. The writer of the raster file will use the native word order, and the reader is responsible for detecting a reversed word order file and swapping bytes as needed. The CUPS Image Library raster functions perform this function automatically.

Following the synchronization word are a series of raster pages. Each page starts with a page device dictionary header and is followed immediately by the raster data for that page.

Bytes	Description	Values
0–63	MediaClass	Nul-terminated ASCII string
64–127	MediaColor	Nul-terminated ASCII string
128–191	MediaType	Nul-terminated ASCII string
192–255	OutputType	Nul-terminated ASCII string
256–259	AdvanceDistance	0 to $2^{32} - 1$ points
260–263	AdvanceMedia	0 = Never advance roll 1 = Advance roll after file 2 = Advance roll after job 3 = Advance roll after set 4 = Advance roll after page
264–267	Collate	0 = do not collate copies 1 = collate copies
268–271	CutMedia	0 = Never cut media 1 = Cut roll after file

## CUPS Interface Design Description

		2 = Cut roll after job 3 = Cut roll after set 4 = Cut roll after page
272–275	Duplex	0 = Print single-sided 1 = Print double-sided
276–283	HWResolution	Horizontal and vertical resolution in dots-per-inch.
284–299	ImagingBoundingBox	Four integers giving the left, bottom, right, and top positions of the page bounding box in points
300–303	InsertSheet	0 = Do not insert separator sheets 1 = Insert separator sheets
304–307	Jog	0 = Do no jog pages 1 = Jog pages after file 2 = Jog pages after job 3 = Jog pages after set
308–311	LeadingEdge	0 = Top edge is first 1 = Right edge is first 2 = Bottom edge is first 3 = Left edge is first
312–319	Margins	Left and bottom origin of image in points
320–323	ManualFeed	0 = Do not manually feed media 1 = Manually feed media
324–327	MediaPosition	Input slot position from 0 to N
328–331	MediaWeight	Media weight in grams per meter squared
332–335	MirrorPrint	0 = Do not mirror prints 1 = Mirror prints
336–339	NegativePrint	0 = Do not invert prints 1 = Invert prints
340–343	NumCopies	1 to $2^{32} - 1$
344–347	Orientation	0 = Do not rotate page 1 = Rotate page counter-clockwise 2 = Turn page upside down 3 = Rotate page clockwise
348–351	OutputFaceUp	0 = Output face down 1 = Output face up
352–359	PageSize	Width and length in points

## CUPS Interface Design Description

360–363	Separations	0 = Print composite image 1 = Print color separations
364–367	TraySwitch	0 = Do not change trays if selected tray is empty 1 = Change trays if selected tray is empty
368–371	Tumble	0 = Do not rotate even pages when duplexing 1 = Rotate even pages when duplexing
372–375	cupsWidth	Width of page image in pixels
376–379	cupsHeight	Height of page image in pixels
380–383	cupsMediaType	Driver-specific 0 to $2^{32} - 1$
384–387	cupsBitsPerColor	1, 2, 4, 8 bits
388–391	cupsBitsPerPixel	1 to 32 bits
392–395	cupsBytesPerLine	1 to $2^{32} - 1$ bytes
396–399	cupsColorOrder	0 = chunky pixels (CMYK CMYK CMYK) 1 = banded pixels (CCC MMM YYK KKK) 2 = planar pixels (CCC... MMM... YYK... KKK...)
400–403	cupsColorSpace	0 = white 1 = RGB 2 = RGBA 3 = black 4 = CMY 5 = YMC 6 = CMYK 7 = YMCK 8 = KCMY 9 = KCMYcm
404–407	cupsCompression	Driver-specific 0 to $2^{32} - 1$
408–411	cupsRowCount	Driver-specific 0 to $2^{32} - 1$
412–415	cupsRowFeed	Driver-specific 0 to $2^{32} - 1$
416–419	cupsRowStep	Driver-specific 0 to $2^{32} - 1$

The MIME type for CUPS Raster files is `application/vnd.cups-raster`.

## 4.6 CUPS Raw Files

Raw files are printer-dependent print files that are in a format suitable to the destination printer (e.g. HP-PCL, HP-RTL, etc.) The MIME type for CUPS Raw files is `application/vnd.cups-raw`.

## 4.7 Internet Printing Protocol

The Internet Printing Protocol and the CUPS extensions to it are described in the CUPS Implementation of IPP document.

## 4.8 Line Printer Daemon Protocol

The Line Printer Daemon (LPD) protocol is described by [RFC 1179: Line Printer Daemon Protocol](#).

The URI method name for LPD is "lpd".

## 4.9 Server Message Block Protocol

The Server Message Block (SMB) and related Common Internet File System (CIFS) protocols are described at <http://anu.samba.org/cifs>.

The URI method name for SMB is "smb". Support for this protocol is provided via the SAMBA `smbpool ( 1 )` program provided with SAMBA 2.0.6 and higher.

## 5 Directories

*/etc/cups*

The scheduler configuration and MIME files reside here.

*/etc/cups/certs*

The authentication certificates reside here.

*/etc/cups/interfaces*

System V interface scripts reside here.

*/etc/cups/ppd*

This directory contains PPD files for each printer.

*/usr/bin*

The `cancel`, `lp`, `lpq`, `lpr`, `lprm`, and `lpstat` commands reside here.

*/usr/lib, /usr/lib32*

The shared libraries (DSOs) reside here.

*/usr/lib/cups/backend*

The backend filters reside here.

*/usr/lib/cups/cgi-bin*

The CGI programs reside here.

*/usr/lib/cups/daemon*

The polling and LPD daemons reside here.

*/usr/lib/cups/filter*

The file filters reside here.

*/usr/sbin*

The `accept`, `cupsd`, `lpadmin`, `lpc`, and `reject` commands reside here.

*/usr/share/cups*

This is the root directory of the CUPS static data.

*/usr/share/cups/charsets*

The character set files reside here.

*/usr/share/cups/data*

The filter data files reside here.

*/usr/share/cups/fonts*

The `psraster` font files reside here.

*/usr/share/cups/model*

The sample PPD files reside here.

*/usr/share/cups/psraster*

The `psraster` data files reside here.

*/usr/share/doc/cups*

The scheduler documentation files reside here.

*/var/log/cups*

The `access_log`, `error_log`, and `page_log` files reside here.

*/var/spool/cups*

This directory contains print job files.



# A Glossary

## A.1 Terms

<i>C</i>	A computer language.
<i>parallel</i>	Sending or receiving data more than 1 bit at a time.
<i>pipe</i>	A one-way communications channel between two programs.
<i>serial</i>	Sending or receiving data 1 bit at a time.
<i>socket</i>	A two-way network communications channel.

## A.2 Acronyms

<i>ASCII</i>	American Standard Code for Information Interchange
<i>CUPS</i>	Common UNIX Printing System
<i>ESC/P</i>	EPSON Standard Code for Printers
<i>FTP</i>	File Transfer Protocol
<i>HP-GL</i>	Hewlett-Packard Graphics Language
<i>HP-PCL</i>	Hewlett-Packard Page Control Language
<i>HP-PJL</i>	Hewlett-Packard Printer Job Language
<i>IETF</i>	Internet Engineering Task Force
<i>IPP</i>	Internet Printing Protocol
<i>ISO</i>	International Standards Organization
<i>LPD</i>	Line Printer Daemon
<i>MIME</i>	Multimedia Internet Mail Exchange
<i>PPD</i>	PostScript Printer Description
<i>SMB</i>	Server Message Block
<i>TFTP</i>	Trivial File Transfer Protocol

